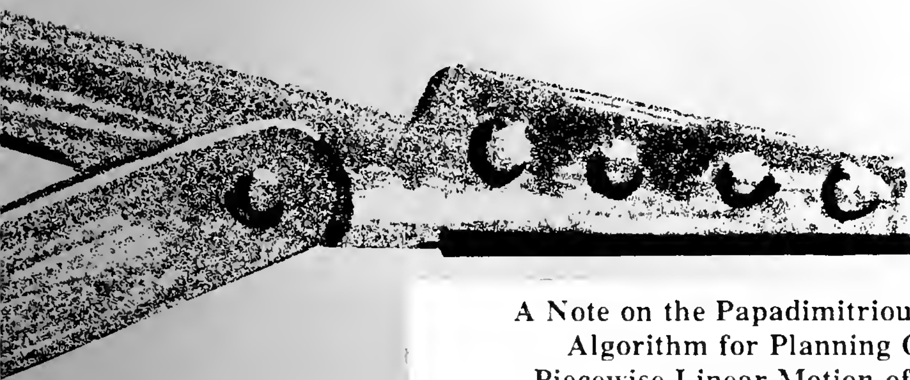
Robotics Research Technical Report

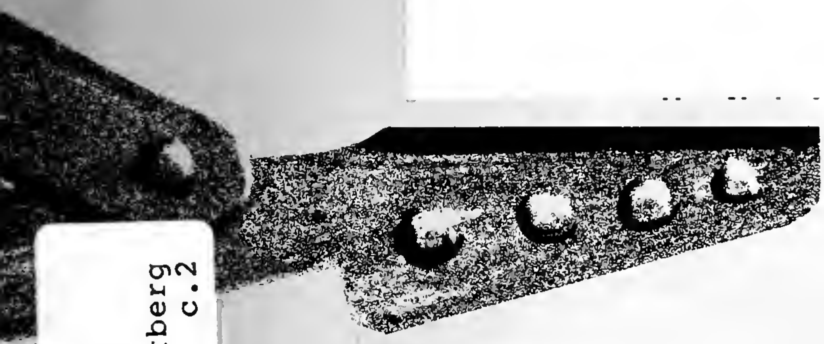


A Note on the Papadimitriou-Silverberg Algorithm for Planning Optimal Piecewise-Linear Motion of a Ladder

by

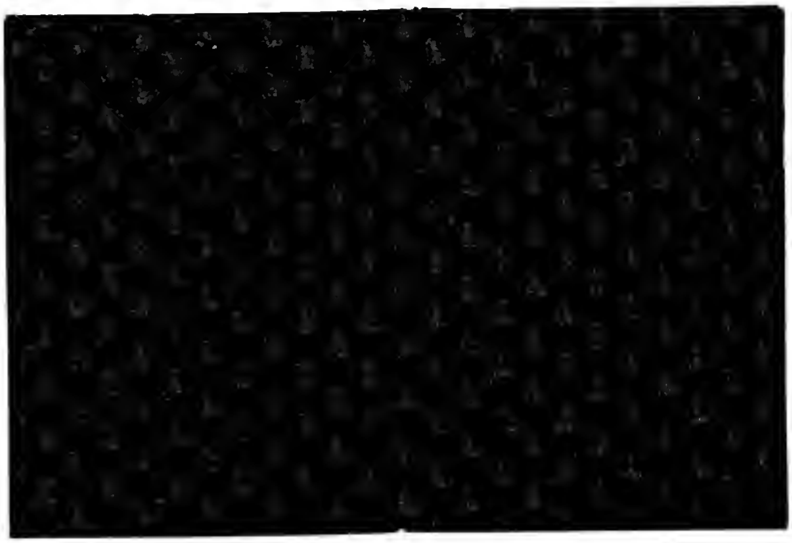
Micha Sharir

Technical Report No. 426
Robotics Report No. 188
January, 1989



New York University
ant Institute of Mathematical Sciences

Computer Science Division
251 Mercer Street New York, N.Y. 10012



A Note on the Papadimitriou-Silverberg Algorithm for Planning Optimal Piecewise-Linear Motion of a Ladder

by

Micha Sharir

Technical Report No. 426
Robotics Report No. 188
January, 1989

New York University
Dept. of Computer Science
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, New York 10012

Work on this paper has been supported in part by Office of Naval Research Grant N00014-87-K-0129, by National Science Foundation Grant No. NSF-DCR-83-20085, by grants from the Digital Equipment Corporation, and the IBM Corporation, and by grants from the U.S.-Israeli Binational Science Foundation, the Israeli National Council for Research and Development, and the EMET Fund of the Israeli Academy of Sciences.

A Note on the Papadimitriou-Silverberg Algorithm for Planning Optimal Piecewise-Linear Motion of a Ladder

Micha Sharir[1,2]

[1]Courant Institute of Mathematical Sciences,
New York University, New York NY 10012

[2]School of Mathematical Sciences,
Tel Aviv University, Israel

[0]Work on this paper has been supported in part by Office of Naval Research Grant N00014-87-K-0129, by National Science Foundation Grant No. NSF-DCR-83-20085, by grants from the Digital Equipment Corporation, and the IBM Corporation, and by grants from the U.S.-Israeli Binational Science Foundation, the Israeli National Council for Research and Development, and the EMET Fund of the Israeli Academy of Sciences.

Abstract

We present an improvement, by nearly an order of magnitude, of a recent algorithm by Papadimitriou and Silverberg for optimal motion planning for a ladder [PS]. The improved algorithm runs in time $O(n^3\alpha(n)\log^2 n)$, where n is the number of obstacle edges and where $\alpha(n)$ is the extremely slowly growing inverse Ackermann function.

In [PS], Papadimitriou and Silverberg consider the following restricted version of optimal motion planning. Suppose we are given a line segment $B = PQ$ (a "ladder") free to move in a two-dimensional closed polygonal region V bounded by a total of n edges. Each placement of B can be specified by (x, θ), where x is the position of P and θ is the orientation of $\vec{PQ}$. A placement (x, θ) is *free* if at this placement B is fully contained in V. Let (s, θ_s), (t, θ_t) be two given initial and final free placements of B. The goal is to plan a collision-free motion of B from (s, θ_s) to (t, θ_t) having the following properties:

(i) The endpoint P moves along straight segments connecting pairs of corners of V and not intersecting V^c (such pairs are said to be *visible* from one another: s and t are also considered as corners in this context).

(ii) The total distance traveled by P is minimum among all collision-free motions from (s, θ_s) to (t, θ_t) satisfying (i).

The algorithm presented in [PS] runs in time $O(n^4 \log n)$. It proceeds roughly as follows. It first computes the *visibility graph* VG of the given obstacle corners (including s and t) in time $O(n^2)$ (the graph connects by edges all pairs of corners that are visible from one another). Next, for each edge $e = uv$ of VG, it spends $O(n^2 \log n)$ time to compute all possible free motions in which P moves along e from u to v, and associates an appropriate cost with each such motion. Combining the transitions along each edge of VG yields an overall "connectivity graph" and the desired optimal motion can then be obtained by applying Dijkstra's algorithm to the connectivity graph.

In this note we show that the step of processing a single edge of VG can be performed in only $O(n\alpha(n) \log^2 n)$, where $\alpha(n)$ is the extremely slowly-growing functional inverse of Ackermann's function. This leads to an overall $O(n^3\alpha(n) \log^2 n)$ algorithm, improving the result of [PS] by nearly an order of magnitude. To obtain this improvement we use tools that have recently been developed in [GSS], [EGPPSS] and [Si] for the analysis and calculation of a single face (or certain collections of faces) in arrangements of curves in the plane. One of the purposes of this note is to demonstrate the significance and broad applicability of these tools, although the application at hand is not completely routine. We suspect that a scan through the literature would yield many other problems whose current solutions can be improved by using this new machinery.

To explain how this improvement is obtained, we first review in more detail the machinery used in [PS] to analyze and solve the subproblem of moving P along an edge $e = uv$ of the visibility graph. This restricted motion has only two degrees of

freedom, and any (free or non-free) placement of B with P on e can be specified by the two parameters (x, θ), where x is the length of uP and θ is the orientation of B (for simplicity, and without loss of generality, let us assume that e lies on the x-axis). The parametric space of all these placements is a cylinder $C_e = \{(x, \theta) : 0 \le x \le L, 0 \le \theta \le 2\pi\}$, where $L = |uv|$.

Each obstacle edge or corner o induces a *contact curve* γ_o in C_e that is the locus of all placements (x, θ) of B in C_e at which its endpoint Q touches o, if o is an edge, or at which B passes through o, if o is a corner. Let $\mathcal{A}$ denote the *arrangement* of the contact curves in C_e. This is the cylindrical map obtained by drawing all these curves; its vertices are endpoints or intersection points of contact curves, its edges are maximal connected portions of the contact curves not containing a vertex, and its faces are maximal connected cells not meeting any contact curve. Any motion of B within C_e which starts at some free placement z and remains free cannot cross any of the contact curves γ_o and thus must remain within the face f of $\mathcal{A}$ containing z. Conversely, any placement in f is clearly reachable from z by a collision-free motion of B within C_e. See [GSS] for more details.

To apply this crucial observation, consider all free placements of B with P touching u (within C_e this is the circle $x = 0$). As noted in [PS], the orientations of these placements fall into $O(n)$ circular arcs, each of which is delimited on both sides by a placement that is an intersection of some contact curve with $x = 0$. The algorithmic step that concerns us is to determine, for each one of these free arcs, δ, which free arcs at $x = L$ are reachable within C_e from δ (more precisely, from $\{0\} \times \delta$), and what is the cost of reaching each of them. (Recall that this cost is the total distance traveled by P; if P does not backtrack during the motion, the cost is simply L, but when backtracking is allowed the cost can be higher.)

The preceding observations therefore suggest the following approach.

(a) Compute all the faces of $\mathcal{A}$ that meet the circle $x = 0$ (this collection is known as the *zone* of $x = 0$ [Ed], [EGPPSS]).

(b) Decompose each such face f into x-monotone regions by drawing, from each point on ∂f that is locally x-extremal, vertical segments that are extended within f until they meet again ∂f.

(c) Build a connectivity graph whose nodes are all these regions and whose edges connect pairs of adjacent regions (along a common vertical segment). Associate with each region a cost equal to its x-width.

(d) Combine all the connectivity graphs, over all edges e of VG, into a single overall connectivity graph CG. Then search for the minimum weight path within CG

between the initial and final placements of B.

The first step can be accomplished by adapting the algorithm of [GSS] (see also [EGPPSS]) for the calculation of a single face in an arrangement of curves. In step (a) we need to calculate more than a single face, but all the desired faces belong to a single zone, so that, using the techniques of [EGPPSS], we can regard all of them as a single face. (Intuitively, we regard C_ϵ as an infinite cylinder, but clip the curves γ_o so that they do not extend to $x < 0$; this makes all the desired faces portions of a common single unbounded face. Note also that the analysis of [GSS], [EGPPSS] is normally carried out for planar arrangements; however, as discussed in [GSS], it is easy to cut our cylindrical arrangement into two planar arrangements, carry out the analysis in each planar patch separately, and then glue together the resulting faces to obtain the desired final faces.)

The analysis of [GSS], [EGPPSS] shows that the total combinatorial complexity of all the faces in question is $O(\lambda_{s+2}(n))$, where s is the maximum number of intersections between any pair of contact curves (we can actually use a slightly smaller s – see below), and where $\lambda_{s+2}(n)$ is the maximum length of (n, s) *Davenport-Schinzel sequences* [HS], [ASS], which is almost linear in n for any fixed s. Moreover, it is shown in [GSS] that all these faces can be computed in total time $O(\lambda_{s+2}(n)\log^2 n)$.

Before continuing with the analysis of the remaining steps, let us analyze the value of s. In a naive approach, the maximum number of intersections between any pair of contact curves is easily seen to be 2. However, we can reduce this number to 1 by applying a recent observation of [Si]. Specifically, we partition the collection of contact curves into three subcollections. The first subcollection Γ_1 consists of all curves γ_o where o is a corner. The second subcollection Γ_2 consists of the portions of all contact curves γ_o where o is an edge, which lie in the range $-\frac{\pi}{2} \le \theta \le \frac{\pi}{2}$, and the third subcollection Γ_3 consists of the portions of the latter curves which lie is the range $\frac{\pi}{2} \le \theta \le \frac{3\pi}{2}$. It is now straightforward to verify the following

Claim 1 *Any pair of curves in the same subcollection Γ_j intersect at most once.*

This claim implies that if we form the arrangement $\mathcal{A}_j$ of the curves in Γ_j then the complexity of the zone of $x = 0$ in $\mathcal{A}_j$ is only $O(\lambda_3(n)) = O(n\alpha(n))$ [HS], and it can be calculated in time $O(n\alpha(n)\log^2 n)$. Then the combination lemma of [GSS] implies that if we superimpose the three arrangements $\mathcal{A}_j$ on each other, the complexity of the resulting zone of $x = 0$ remains $O(n\alpha(n))$, and it can be computed from the three corresponding zones in the $\mathcal{A}_j$'s in additional time $O(n\alpha(n)\log n)$.

Continuing the analysis of the algorithm, we note that step (b) can be done in time $O(n\alpha(n)\log n)$, using a simple line sweeping technique (as a matter of fact, the

algorithm in [GSS] produces this vertical decomposition for free). Step (c) is also trivial to accomplish, in time $O(n\alpha(n))$. Thus the complexity of the overall graph CG is $O(n^3\alpha(n))$, and it can be searched within the same time bound. Thus the overall cost of step (a) dominates all the other steps, and we can conclude

Theorem 1 *One can plan optimal restricted motion of a ladder as above in time* $O(n^3\alpha(n)\log^2 n)$.

We conclude with a few open problems. First, [PS] also considers optimal motion planning where the cost of a path is a linear combination of the distance traveled by P and the total amount of rotation executed by B. For this cost measure, [PS] presents an $O(n^5)$ algorithm, and it would be nice to improve this complexity using our techniques. A second problem is to extend our techniques to the case (not studied in [PS]) where the motion of P is unrestricted (and the cost is just the distance traveled by P).

References

[ASS] P. Agarwal, M. Sharir and P. Shor, Sharp upper and lower bounds on the length of general Davenport-Schinzel sequences, Tech. Rept. 332, Dept. of Computer Science, New York University, 1987.

[Ed] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, Springer Verlag, Heidelberg 1987.

[EGPPSS] H. Edelsbrunner, L. Guibas, J. Pach, R. Pollack, R. Seidel and M. Sharir, Arrangements of curves in the plane: Combinatorics, topology, and algorithms, *Proc. 15th Int. Colloq. on Automata, Languages and Programming*, 1988, pp. 214-229.

[GSS] L. Guibas, M. Sharir and S. Sifrony, On the general motion planning problem with two degrees of freedom, *Proc. 4th ACM Symposium on Computational Geometry*, 1988, pp. 289-298.

[HS] S. Hart and M. Sharir, Nonlinearity of Davenport-Schinzel sequences and of generalized path compression schemes, *Combinatorica* 6 (1986) pp. 151-177.

[PS] C. Papadimitriou and E. Silverberg, Optimal piecewise linear motion of an object among obstacles, *Algorithmica* 2 (1987) pp. 523-539.

[Si] S. SIFRONY, Ph.D. Dissertation, in preparation.

NYU COMPSCI TR-426
Sharir, Micha
A note on the
Papadimitriou-Silverberg
algorithm for... c.2